UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/903,100	07/11/2001	Ramesh Subramanian	GSH 08-885923	1887

27667	7590	05/16/2006

HAYES, SOLOWAY P.C.
3450 E. SUNRISE DRIVE, SUITE 140
TUCSON, AZ 85718

EXAMINER
CAO, DIEM K

ART UNIT	PAPER NUMBER
2194	

DATE MAILED: 05/16/2006

Please find below and/or attached an Office communication concerning this application or proceeding.

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE <u>3</u> MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.
- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on <u>20 March 2006</u>.

2a)☐ This action is **FINAL**.   2b)☒ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) <u>1,6,9-13,17-21,23,24 and 28-34</u> is/are pending in the application.

   4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) <u>1,6,9-13,17-21,23,24 and 28-34</u> is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

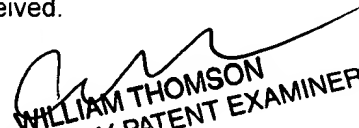8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☐ The drawing(s) filed on _____ is/are: a)☐ accepted or b)☐ objected to by the Examiner.

   Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

   Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

   a)☐ All b)☐ Some * c)☐ None of:

   1.☐ Certified copies of the priority documents have been received.

   2.☐ Certified copies of the priority documents have been received in Application No. _____.

   3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

   * See the attached detailed Office action for a list of the certified copies not received.

WILLIAM THOMSON
SUPERVISORY PATENT EXAMINER

Attachment(s)

1)☐ Notice of References Cited (PTO-892)

2)☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)

3)☐ Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08) Paper No(s)/Mail Date _____.

4)☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____.

5)☐ Notice of Informal Patent Application (PTO-152)

6)☐ Other: _____.

DETAILED ACTION

1.   Claims 1, 6, 9-13, 17-21, 23, 24 and 28-34 are pending. Applicant has amended claims 1,

6, 9, 13, 17, 21, 24, 28, 33 and 34 and canceled claims 2-5, 7-8, 14-16, 22 and 25-27.

Continued Examination Under 37 CFR 1.114

2.   A request for continued examination under 37 CFR 1.114, including the fee set forth in

37 CFR 1.17(e), was filed in this application after final rejection. Since this application is

eligible for continued examination under 37 CFR 1.114, and the fee set forth in 37 CFR 1.17(e)

has been timely paid, the finality of the previous Office action has been withdrawn pursuant to

37 CFR 1.114. Applicant's submission filed on 3/20/2006 has been entered.

Claim Rejections - 35 USC § 101

3.   35 U.S.C. 101 reads as follows:

> Whoever invents or discovers any new and useful process, machine, manufacture, or composition of matter, or
> any new and useful improvement thereof, may obtain a patent therefor, subject to the conditions and
> requirements of this title.

Claim 34 is rejected under 35 U.S.C. 101 because the claimed invention is directed to

non-statutory subject matter. The claims are directed to a signal directly or indirectly by

claiming a medium and the Specification recite evidence where the computer readable

medium is define as a *"wave"* (such as a carrier wave). In that event, the claims are directed

to a form of energy which at present the office feels does not fall into a category of invention.

The following link on the World Wide Web is for the United States Patent And Trademark

Office (USPTO) policy on 35 U.S.C. §101.

<http://www.uspto.gov/web/offices/pac/dapp/opla/preognotice/guidelines101_20051026.pdf>

Claim Rejections - 35 USC § 103

4.   The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all

obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

5.   **Claims 1, 6, 13, 21, 23, 24, 28 and 33-34 are rejected under 35 U.S.C. 103(a) as being**

unpatentable over Kannan et al. (U.S. 5,815,702) in view of Bak et al. (U.S. 6,415,381 B1)

further in view of Anschuetz et al. (U.S. 5,305,455).

6.   As to claim 1, Kannan teaches receiving an exception caused due to a runtime fault in a

thread executing the application (The current instruction 302 ... a general protection fault; col. 6,

line 66 – col. 7, line 2), dispatching the exception to an exception handler (The operating system

... to the chain of exception handlers; col. 7, lines 5-8), trapping the exception before the

exception reaches the exception handler when the exception handler is a top level exception

handler which terminates the application (The exception handler 115 ... terminate the

application; col. 4, lines 44-47 and col. 1, line 48 – col. 2, line 11), and continuing execution of

the application (the application to continue executing; col. 4, lines 58-64 and the exception

handler ... in the proper manner; col. 7, lines 29-33).

7.   However, Kannan does not teach translating the trapped exception into an exception that

the application is capable of handling, the application is a C++ based application and the

exception is a C++ exception, C++ exception handler, determining if there is an application

based C++ exception handler which is capable of resolving the translated exception, and

terminating the thread that caused the exception when there is not C++ based exception handler

which is capable of resolving the translated exception.

8.   Bak teaches a Java based application (a Java program; col. 5, lines 1-2), and translating

the trapped exception into an exception that the application is capable of handling, wherein the

trapped exception is translated into the exception which is able to be resolved by an application

exception handler (Shield 715 catches ...the exception is passed on; col. 11, lines 50-55),

determining if there is an application exception handler which is capable of resolving the

translated exception (col. 12, lines 46-63). Although Bak does not teach C++ language, one of

ordinary skill in the art could apply the concept taught by Bak into different programming

language. Furthermore, both Java and C++ are object-oriented programming languages. It would

have been obvious to one of ordinary skill in the art at the time the invention was made to

combine the teaching of Kannan and Bak because it would improve the performance the system

of Kannan by being able to handle the exception in multiple programming languages.

9.   Anschuetz teaches terminating the thread that caused the exception (col. 5, lines 9-16). It

would have been obvious to one of ordinary skill in the art at the time the invention was made to

combine the teaching of Kannan and Anschuetz because it provides a method to handle the exception on a per thread basis.

10.   As to claim 6, Kannan does not teach the continuing step allows continuing execution of the application after the thread is terminated. Kannan teaches the application is continue execution after the application generates a fatal exception which always causes termination to the application (abstract). Anschuetz teaches terminating the thread that caused the exception (col. 5, lines 9-16). It would have been obvious to one of ordinary skill in the art at the time the invention was made to combine the teaching of Kannan and Anschuetz because it provides a method to handle the exception on a per thread basis, and the application has change to continue execution.

11.   As to claim 13, see rejection of claim 1 above. Kannan further teaches the application being executed under an operating system having one or more low level exception handlers and a top level exception handler (Loaded into and executing …processor of a computer; col. 4, lines 4-12 and The operating system … itself; col. 4, lines 23-30).

12.   As to claim 21, it corresponds to the method claim of claim 1 except it is a computer system claim.

13.   As to claim 23, Kannan teaches the exception trapper is provided in place of a top level exception handler which terminates the application (The exception handler 115 is inserted in the

exception handler chain 114 ahead of all operating system provided exception handlers; col. 6,

lines 37-42).

14.   As to claim 24, Kannan teaches an application system for recovering an application from

a runtime fault caused in a thread (A method ... a fatal exception; abstract), the application

running under an operating system having an exception dispatcher (inherent from the operating

system ... of exception handlers; col. 7, lines 5-8), one or more low level exception handlers and

a top level exception handler which terminates the application (The operating system ... itself;

col. 4, lines 23-30 and Certain types of application errors ... fatal exception; col. 1, line 47 – col.

2, line 11), an exception trapper placed between the exception dispatcher and the top level

exception handler for trapping an exception before the exception reaches the top level exception

handler (The exception handler 115 is inserted in the exception handler chain 114 ahead of all

operating system provided exception handlers; col. 6, lines 37-42 and The exception handler 115

... terminate the application; col. 4, lines 44-47), the exception being caused due to a runtime

fault in a thread executing an application (The current instruction 302 ... a general protection

fault; col. 6, line 66 – col. 7, line 2), and a trapped exception handler for handling the trapped

exception (The exception handler 115 ... terminate the application; col. 4, lines 44-47 and the

crash guard process 107; col. 4, lines 38-64).

15.   However, Kannan does not teach translating the trapped exception into an exception that

the application is capable of handling, nor the application is a C++ based application, the

exception is a C++ exception, a trapped exception handler for handling the trapped exception,

the trapped exception handler having an exception handler selector for determining if there is an

application based C++ exception handler that is capable of resolving the exception translated by

the exception translator, and a thread terminator for terminating the thread when there is not C++

based exception handler that is capable of handling the translated exception.

16.   Bak teaches a Java based application (a Java program; col. 5, lines 1-2), and translating

the trapped exception into an exception that the application is capable of handling, wherein the

trapped exception is translated into the exception which is able to be resolved by an application

exception handler (Shield 715 catches ...the exception is passed on; col. 11, lines 50-55),

determining if there is an application exception handler which is capable of resolving the

translated exception (col. 12, lines 46-63). Although Bak does not teach C++ language, one of

ordinary skill in the art could apply the concept taught by Bak into different programming

language. Furthermore, both Java and C++ are object-oriented programming languages. It would

have been obvious to one of ordinary skill in the art at the time the invention was made to

combine the teaching of Kannan and Bak because it would improve the performance the system

of Kannan by being able to handle the exception in multiple programming languages.

17.   Anschuetz teaches terminating the thread that caused the exception (col. 5, lines 9-16). It

would have been obvious to one of ordinary skill in the art at the time the invention was made to

combine the teaching of Kannan and Anschuetz because it provides a method to handle the

exception on a per thread basis.

18.   As to claim 28, Kannan teaches the trapped exception handler further comprises a state

restorer for restoring the state that the application was in before the fault occurred to continued

the execution of the application (col. 7, lines 3-5).

19.   As to claims 33 and 34, they correspond to the method claim of claim 1 except they are

computer readable memory element of computer electronic signals claims, respectively.

20.   Claims 9-12, 17-20, and 29 are rejected under 35 U.S.C. 103(a) as being

unpatentable over Kannan et al. (U.S. 5,815,702) in view of Bak et al. (U.S. 6,415,381 B1)

and Anschuetz et al. (U.S. 5,305,455) further in view of LeVine et al. (U.S. 6,591,379 B1).

21.   As to claim 9, Kannan does not teach logging state information representing the state that

the application was in before occurrence of the exception caused the termination of the thread.

LeVine teaches logging state information representing the state that the application was in before

occurrence of the exception caused the termination of the thread (col. 7, lines 2-7 and col. 8,

lines 41-46 and Fig. 5). It would have been obvious to one of ordinary skill in the art at the time

the invention was made to combine the teaching of Kannan and LeVine because LeVine's

logging state information would improve the reliability of Kannan's system by saving all

necessary information for later recovery.

22.   As to claim 10, Kannan does not teach forwarding the logged information to a remote

database over a computer network. LeVine teaches forwarding the logged information to a

remote database over a computer network (col. 7, lines 5-8).

23.   As to claim 11, Kannan teaches receiving a recommendation from the remote database

(col. 6, lines 15-18), and informing the recommendation to the user (col. 7, lines 34-44).

24.   As to claim 12, Kannan does not teach forwarding a bug report to a bug report center

over a computer network. LeVine teaches forwarding a bug report to a bug report center over a

computer network (col. 8, lines 1-8).

25.   As to claim 17, see rejection of claim 9 above.

26.   As to claim 18, see rejection of claim 10 above.

27.   As to claim 19, see rejection of claim 11 above.

28.   As to claim 20, see rejection of claim 12 above.

29.   As to claim 29, see rejection of claim 9 above.

30. **Claims 30-32 are rejected under 35 U.S.C. 103(a) as being unpatentable over Kannan et al. (U.S. 5,815,702) in view of Bak et al. (U.S. 6,415,381 B1), Anschuetz et al. (U.S. 5,305,455) and LeVine et al. (U.S. 6,591,379 B1) further in view of Lillevold (U.S. 6,230,284 B1).**

31.   As to claim 30, Kannan does not teach a query generator for generating a query including the state information to query a recommendation from a remote database over a computer network. Lillevold teaches the crash handler program determines the state of the computer, sends the information to the server, and the server, based on this information, send revision code to the computer to avoid the crash next time (col. 3, lines 21-27). It would have been obvious to one of ordinary skill in the art at the time the invention was made to combine the teaching of Kannan and Lillevold because it would improve the performance of Kannan system by obtaining correct help from server when it is not available locally.

32.   As to claim 31, see rejection of claim 11 above.

33.   As to claim 32, see rejection of claim 12 above.

Response to Arguments

34.   Applicant's arguments filed 3/20/2006 have been fully considered but they are not persuasive.

Applicant arguments in substance that Anschuetz does not disclose exception handling at the application level because Anschuetz receives an exception of an operation system, and

dispatches the exception to an exception handler, and thread in the system is Anschuetz is a

thread executing a process in the operating system, and Anschuetz's system manages exceptions

on a thread basis in the operating system, wherein the present invention as recited in claim 1

enables the C++ based application to handle exceptions by itself programmatically.

Examiner respectfully traverses Applicant's arguments:

1) The claims are rejected under the combination of Kannan, Bak, and Anschuetz, not

any one of them alone. As set forth in the rejection of claim 1, Kannan and Bak clearly teach the

application is able to handle exceptions by itself programmatically.

2) The exception in Anschuetz is application based exception (user exception; col. 4,

lines 1-3, 33).

3) The reference of Anschuetz is used to teach terminating the thread that causes the

exception. Thus, one of ordinary skill in the art would be motivated to apply the teaching of

Anschuetz to the system ofKannan.

Therefore, the arguments are not persuasive.

Conclusion

Any inquiry concerning this communication or earlier communications from the

examiner should be directed to Diem K. Cao whose telephone number is (571) 272-3760. The

examiner can normally be reached on Monday - Friday, 7:30AM - 3:30PM.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's

supervisor, William Thomson can be reached on (571) 272-3718. The fax phone number for the

organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent

Application Information Retrieval (PAIR) system. Status information for published applications

may be obtained from either Private PAIR or Public PAIR. Status information for unpublished

applications is available through Private PAIR only. For more information about the PAIR

system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR

system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

Any response to this action should be mailed to:
Commissioner for Patents
PO Box 1450
Alexandria, VA 22313-1450

Diem Cao

WILLIAM THOMSON
SUPERVISORY PATENT EXAMINER